**Exercise 1: Implementing the Singleton Pattern**

**Scenario:**

You need to ensure that a logging utility class in your application has only one instance throughout the application lifecycle to ensure consistent logging.

**Code:**

class Logger{

    static Logger abc = new Logger();

    private Logger(){

    }

    public static Logger getInstance(){

        System.out.println("instance created");

        return abc;

    }

    public void display(){

        System.out.println("Singleton pattern implementedd");

    }

}

public class SingletonPatternExample{

    public static void main(String[] args) {

        Logger a = Logger.getInstance();

        Logger b = Logger.getInstance();

        // a.display();

        if(a==b){

            System.out.println("both instances are same");

        }

        else{

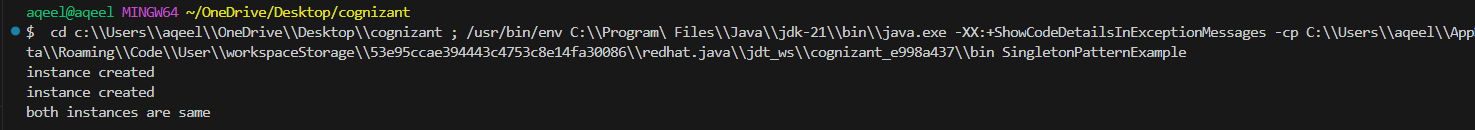
            System.out.println("Both instances are different");

        }

    }

}

**OUTPUT:**

****